

一、填空

1. break 语句通常用于 switch // 循环 中。
2. C 语言对数组下标的引用一般从 0 开始。
3. C 语言中，一个函数一般由两个部分组成，它们是 函数首部 和 函数体。
4. C 标准库函数中，字符串的处理函数包含在 string.h 头文件中，数学函数包含在 math.h 头文件中。
5. C 程序的运行需要经过 编译 和 链接 两步进行。
6. C 程序中用 /**/ 括起来的内容是程序的注释语句。
7. C 语言函数是由函数首部和函数体两部分组成。其中，函数首部包括 函数类型、函数名 和 函数参数。
8. C 语言提供的三种逻辑运算符是 &, //, !。
9. C 语言源程序经过编译后，生成文件的后缀名是 .c。
10. C 语言源程序经过连接后，生成文件的后缀名是 .exe。
11. C 语言中，关系表达式和逻辑表达式的值是 1//0。
12. C 语言中的标识符只能由三种字符组成，他们是 字母，数字，下划线。
13. C 语言中的每条基本语句以 _____ 作为结束符，每条复合语句以 } 作为结束符。
14. C 语言中函数返回值的类型是由 函数类型 决定的。
15. C 源程序的基本单位是 函数。
16. `int a = 2, b = 3;` 表达式 `a > b ? a++ : ++b` 的值为 4。
17. `int x=1,y=1,z=1; x+=y+=z;` 则表达式 `x<y?y:x` 的值为 3。
18. 当 `a=3` , `b=2` , `c=1` 时，表达式 `f=a>b>c` 的值是 0。
19. 定义一个指针 `p`，它指向一个有 6 个整型元素的一维数组，定义语句为 `int *p=a[6];`。
20. 定义一个指针数组 `p`，它有 4 个元素，每个指针指向一个整型数据，定义语句为 `int *p[4];`。
21. 赋值语句是由 `=` 加上一个分号构成。
22. 构成数组的各个元素必须具有相同的 类型。
23. 关系表达式 `x+y>5` 的相反表达式为 `x+y ! >5`。
24. 函数 `strlen("1234\0xy")` 的值为：9。
25. 函数的形参在未被调用前不分配空间，函数形参的数据类型要和实参相同。
26. 函数体用 `{` 开始，用 `}` 结束。
27. 结构化设计中的三种基本结构是 顺序，选择，循环。

28. 如果一个函数无返回值，定义它时函数类型应是 void。
29. 若 x、y、z 均为 int 型变量，则描述“x 或 y 中有一个小于 z”的表达式是 (x||y)<z。
- 若用数组名作为函数调用时的实参，则实际上传递给形参的是 数组的首地址。
30. 。
31. 若有定义：double x[3][5] ；则数组中列下标的上限为 4。
32. 若有说明：int a[][3]={1,2,3},{4,5},{6,7}; 则数组 a 的第一维的大小为 3。
33. 设 int a=12，则表达式 a/=a+a 运算后，a 的值是 0。
34. 设 y 为 int 型变量，请写出描述“y 是奇数”的表达式 y%2!=0;。
35. 设有数组 a[10]，数组元素 a[3] 的地址可以写成 。
36. 数组由数组名和下标构成，其数组元素的数据类型是 。
37. 算术表达式 a=b+3 中，a、b 称为 变量，3 称为 常量，=、+ 称为 运算符。
38. 一个 C 程序有且仅有一个 main 函数。
39. 已知函数定义为：int stat (int a, float *p)，则此函数的两个参数类型分别是 整形、浮点型。
40. 有如下函数头：int max(int i,int j) ，则此函数有 2 个参数。
41. 预处理命令以 # 开头。
42. 在 C 语言的表达式中，x=x+5 用复合运算符可以写成 x+=5。
43. 在 C 语言中，表示静态存储类别的关键字 static。
44. 在 C 语言中，二维数组元素的内存中的存放顺序是 行顺序存放。
45. 在 C 语言中，以 '\0' 作为字符串结束标志。
46. 在 C 语言中的实型变量分为两种类型，它们是 整形 和 浮点型。
47. 在 scanf 函数中，用符号 & 表示变量的地址。
48. 在传值调用中，实际是把实参的 地址传给形参。
49. 在函数中使用 return 语句返回值。
50. 指针是存放 地址 的变量。

二、选择

- 1、C 语言源程序的基本单位是 A
- A. 函数 B. 标识符 C. 表达式 D. 语句
- 2、在每个 C 程序中都必须包含有这样一个函数，该函数的函数名为 A
- A. main B. MAIN C. name D. function

- 3、C 语言规定，在一个源程序中，main 函数的位置 C
- A. 必须在最开始 B. 必须在系统调用的库函数的后面
- C. 可以任意 D. 必须在最后
- 4、在以下关于 C 语言的注释的叙述中，不正确的是 C
- A. 注释可用 "/*" 、 "*/" 形式表示，也可用 "//" 形式表示
- B. 编译器在编译一个程序时，将跳过注释，不对其进行处理
- C. 编译器在编译一个程序时，可发现注释中的单词拼写错误
- D. 程序中注释的多少不会影响所生成的可执行文件的长度
- 5、以下叙述不正确的是 C
- A. 在 C 程序中，严格区分大小写字母
- B. 一个 C 源程序有且仅有一个 main 函数
- C. 在 C 程序中，注释只能位于一条语句的后面
- D. 一个 C 程序总是从 main 函数开始执行
- 6、C 语言中，表示一条语句结束的标号是 B
- A. # B. ; C. // D. }
- 7、在 C 语言中，以下叙述不正确的是 A
- A) 在 C 程序中，无论是整数还是实数，都能被准确无误的表示
- B) 在 C 程序中，变量名代表存储器中的一个位置
- C) 静态变量的生存期与整个程序的生存期相同
- D) C 语言中变量必须先定义后引用
- 8、C 语言程序编译过程中产生的文件有 B
- A. 索引文件和文本文件两种 B. 目标文件和可执行文件
- C. 二进制文件一种 D. ASCII 码文件和二进制文件两种
- 9、C 语言中的变量名只能由字母，数字和下划线三种字符组成，且第一个字符 C
- A) 必须为字母 B) 必须为下划线
- C) 必须为字母或下划线 D) 可以是字母，数字或下划线中的任意一种
- 10、C 语言中非空的基本数据类型包括 B
- A) 整型，实型，逻辑型 B) 整型，实型，字符型
- C) 整型，字符型，逻辑型 D) 整型，实型，逻辑型，字符型
- 11、下列哪一个表达式的值是整数类型 A
- A. 15/2 B. 15/2+2.0 C. 25/5.0 D. 0.5*10
- 12、设变量 a 是整形，f 是实型，i 是双精度型，则表达式 10+ a +i*f 值的数据类型 C
- A. int B. float C. double D. 不确定
- 13、设有说明 :char w; int x; float y; double z; 则表达式 :w*x+z-y 值的数据类型 D
- A) float B) char C) int D) double
- 14、在 C 语言中，char 型数据在内存中的存储形式是 D
- A. 补码 B. 反码 C. 源码 D. ASCII 码

- 15、下列字符列中，可以作为“字符串常量”的是 B
A. ABC B. xyz C. uvw D. a
- 16、以下运算符中，优先级最高的运算符是 D
A. = B. != C. *(乘号) D. ()
- 17、设有定义 `int k=0;` 则以下四个表达式中与其它三个的值不同的是 C D
A. `k++` B. `k+=1`
C. `++k` D. `k+1`
- 18、设有语句 `scanf("%d,%d ",&m,&n);` 要使 `m`、`n` 的值依次为 2, 3, 正确的输入是 B
A. 2 ?3 B. 2 , 3 C. 2;3 D. 2 3
- 19、已知字母 `A` 的 ASCII 码为十进制数 65, 且 `c2` 为字符型, 则执行语句 `c2 = 'A' + '6' - '3';` 后 `c2` 中的值为 A
A) D B) 68 C) 不确定的值 D) C
- 20、设 `a=6`, `b=4`, 则表达式 `b*=a+3` 的值为 D
A) 3 B) 18 C) 27 D) 36
- 21、若有定义: `int a=8, b=5, c;` 执行语句 `c=a/b+0.4;` 后 `c` 的值为 B
(A) 1.4 (B) 1 (C) 2.0 (D) 2
- 22、若已定义 `x` 和 `y` 是整型变量, `x=2;` 则表达式 `y=2.75+x/2` 的值是 C
A) 5.5 B) 5 C) 3 D) 4.0
- 23、设有以下语句: `int x=10; x+=3+x%(3);` 则 `x` 的值是 A
A) 14 B) 15 C) 11 D) 12
- 24、在 C 语言中逻辑值“真”是用 B 表示的。
A. 预定义为真值的宏 B. 非零值 C. 正数 D. 非零整数
25. C 语言对于嵌套 `if` 语句规定 `else` 总是与 C 匹配。
A) 与最外层的 `if` B.) 与之前最近的 `if`
C) 与之前最近的不带 `else` 的 `if` D) 与最近的 `{ }` 之前的 `if`
- 26、为了避免嵌套的 `if-else` 语句的二义性, C 语言规定 `else` 总是与 C 组成配对关系。
A) 缩排位置相同的 `if` B) 在其之前未配对的 `if`
C) 在其之前尚未配对的最近的 `if` D) 同一行上的 `if`
- 27、设 `x` 和 `y` 均为逻辑型变量, 则 `x && y` 为真的条件是 A
A. 它们均为真 B. 其中一个为真
C. 它们均为假 D. 其中一个为假
- 28、能正确表示逻辑关系: "`a` 10 或 `a` 0" 的 C 语言表达式是 D
A. `a>=10 or a<=0` B. `a>=0|a<=10`
C. `a>=10&&a<=0` D. `a>=10||a<=0`
- 29、判断 `char` 型变量 `c1` 是否为小写字母的正确表达式为 D
A. `'a' <= c1 <= 'z'` B. `(c1 >= a)&&(c1 <= z)`

C. (' a ' > =c1) (' z ' < = c1) D. (c1 > = ' a ')&&(c1 < = ' z ')

30、设 ch 是 char 型变量,值为 'A', 则表达式 ch=(ch>='A' && ch<='Z')?ch+32:ch 的值是 B

A) Z B) a C) z D) A

31、表示关系 $X \leq Y \leq Z$ 的 C 语言表达式为 A

A. (X < =Y)&&(Y < =Z) B. (X < =Y)AND(Y < =Z)
C. (X < =Y < =Z) D. (X < =Y)&(Y < =Z)

32、如果 a=4,b=3,c=2,d=1, 则条件表达式 a < b?a:c < d?c:d 的值是 A

A) 1 B) 2 C) 3 D) 4

33、语句 printf("%d \n",12&&12); 的输出结果是 B

A. 12 B. 1 C. 24 D. 0

34、若有条件表达式 (exp)?a++:b--, 则以下表达式中能完全等价于表达式 (exp) 的是 B

A. (exp==0) B. (exp!=0) C. (exp==1) D. (exp!=1)

35、若要求在 if 后一对圆括号中表示 a 不等于 0 的关系,则能正确表示这一关系的表达式 D

A) a<>0 B) !a C) a=0 D) a

36、设 x、y、z、t 均为 int 型变量,则执行以下语句后,t 的值为 C

x=y=z=1;

t=++x || ++y && ++z;

A) 不定值 B) 4 C) 1 D) 0

37、对表达式 for(表达式 1;; 表达式 3) 可理解为 B

A.)for(表达式 1;0; 表达式 3) B)for(表达式 1;1; 表达式 3)
C)for(表达式 1; 表达式 1; 表达式 3) D)for(表达式 1; 表达式 3; 表达式 3)

38、语句 while(!e); 中的条件 !e 等价于 A

A) e==0 B) e!=1 C) e!=0 D) ~e

39、下面程序段的运行结果是 D

for(i=1;i<=5;)

printf("%d",i);

i++;

A) 12345 B) 1234 C) 15 D) 无限循环

40、设执行变量 y 值为 3, 执行下列循环语句后, 变量 y 的值是 C

do y++; while(y++<4);

(A)3 (B)4 (C)5 (D)6

41、执行语句: for(i=1;i<9; i++) 后; 变量 i 的值是 B

(A) 10 (B) 9 (C) 不定 (D) 11

42、执行语句: for(i=1;i<=9; i++); 后变量 i 的值是 A

A. 10 B. 9 C. 不定 D. 11

- 43、在定义一个一维数组时，不能用来表示数组长度的是 D
A) 常量 B) 符号常量 C) 常量表达式 D) 已被赋值的变量
- 44、判断字符串是否相等，应当使用 D
(A) if(a==b) (B) if(a=b) (C) if(strcpy(a,b)) (D) if(!strcmp(a,b))
- 45、执行程序段 `char str[20]='a','b','c','\0'; int len=strlen(str);` 变量 len 的值 C
A. 20 B. 4 C. 3 D. 3 或 4 或 20
- 46、若有说明 `:char c[10]='E','a','s','t','\0';`，则下述说法中正确的是 D
A) c[7] 不可引用 B) c[6] 可引用，但值不确定
C) c[4] 不可引用 D) c[4] 可引用，其值为空字符
- 47、运行下面程序段的输出结果是 A
`char s1[10]='S','e','t','\0','u','p','\0';
printf("%s",s1);`
A) Set B) Setup C) Set up D) 'S'e't'
- 48、以下程序段的输出结果是 B
`char s[]="an apple";
printf("%d\n",strlen(s));`
A) 7 B) 8 C) 9 D) 10
- 49、下列一维数组初始化语句中，正确且与语句 `float a[]={0,3,8,0,9};` 等价的是 D
A) `float a[6]={0,3,8,0,9};` B) `float a[4]={0,3,8,0,9};`
C) `float a[7]={0,3,8,0,9};` D) `float a[5]={0,3,8,0,9};`
- 50、如有定义语句 `int a[]={1,8,2,8,3,8,4,8,5,8};`，则数组 a 的大小是 A
A) 10 B) 11 C) 8 D) 不定
- 51、若二维数组 a 有 m 列，则在 `a[i][j]` 前面的元素个数为 B
A. $j*m+i$ B. $i*m+j$ C. $i*m+j-1$ D. $i*m+j+1$
- 52、在定义 `int a[5][6];` 后，数组 a 中的第 10 个元素是 C
A) `a[2][5]` B) `a[2][4]` C) `a[1][3]` D) `a[1][5]`
- 53、以下二维数组 a 的正确说明是 C
A) `int a[1,1];` B) `float a(3,4);` C) `double a[1][4];` D) `float a(3)(4);`
- 54、在定义 `int a[5][4];` 之后，对 a 的引用正确的是 C
A) `a[2][4]` B) `a[1,3]` C) `a[4][3]` D) `a[5][0]`
- 55、若有说明：`int a[3][4];` 则对 a 数组元素的正确引用是 C
(A) `a[2][4]` (B) `a[1, 3]` (C) `a[1+1][0]` (D) `a(2)(1)`
- 56、以下对二维数组 a 的正确说明是 C
A. `int a[3][]` B. `float a(3,4)` C. `double a[1][4]` D. `float a(3)(4)`
- 57、若有定义：`int b[3];` 则数组元素的不合法引用是 D
(A) `b[1]` (B) `b[0]` (C) `b[2]` (D) `b[3]`

- 58、对以下说明语句 `int a[10]={6,7,8,9,10};` 的正确理解是 B
- A) 将 5 个初值依次赋给 `a[1]` 至 `a[5]` B) 将 5 个初值依次赋给 `a[0]` 至 `a[4]`
C) 将 5 个初值依次赋给 `a[6]` 至 `a[10]` D) 因为数组长度与初值的个数不相同，所以此语句不正确
- 59、设已定义：`int x[2][4]={1,2,3,4,5,6,7,8}`；则元素 `x[1][1]` 的正确初值是 A
- A. 6 B. 5 C. 7 D. 1
- 60、在 `int a[][3]={ {1},{3,2},{4,5,6},{0}}` 中 `a[1][1]` 的值是 D
- A、0 B、6 C、5 D、2
- 61、设已定义：`int x[2][4]={1,2,3,4,5,6,7,8}`；则元素 6 对应的是 A
- A. `x[1][1]` B. `x[1][2]` C. `x[2][1]` D. `x[2][2]`
- 62、以下正确的函数定义形式是 A
- A. `double fun(int x, int y)` B. `double fun(int x ;int y)`
C. `double fun(int x, int y);` D. `double fun(int x,y);`
- 63、有如下函数调用语句 `func(rec1,rec2+rec3,rec4,rec5);` 该函数调用语句中，含有的实参个数是 B
- A) 3 B) 4 C) 5 D) 有语法错误
- 64、在 C 语言中，全局变量的存储类别是 A
- A) `static` B) `extern` C) `void` D) `register`
- 65、在 C 语言中，以下说法中正确的是 A
- A. 实参与其对应的形参各占用独立的存储单元
B. 实参与其对应的形参占用同一个存储单元
C. 只有当实参与形参同名时才占用同一个存储单元
D. 实参占用存储单元，但形参是虚拟的，不占用存储单元
- 66、C 语言中，简单变量做实参时，它和对应的形参之间的数据传递方式是 B
- (A) 地址传递 (B) 单向值传递
(C) 由实参传给形参，再由形参传递给实参
(D) 同用户指定传递方式
- 67、在一个源文件中定义的全局变量的作用域为 B
- A. 本文件的全部范围 B. 从定义该变量开始至本文件结束
C. 本函数的全部范围 D. 本程序的全部范围
- 68、如果一个函数位于 C 程序文件的上部，在该函数体内说明语句后的复合语句中定义了一个变量，则该变量 C
- A) 为全局变量，在本程序文件范围内有效
B) 为局部变量，只在该函数内有效
C) 为局部变量，只在该复合语句中有效

D) 定义无效，为非法变量

69、调用函数时，当实参和形参都是简单变量时，它们之间数据传递的过程是 D

- A) 实参将其地址传递给形参，并释放原先占用的存储单元
- B) 实参将其地址传递给形参，调用结束时形参再将其地址回传给实参
- C) 实参将其值传递给形参，调用结束时形参再将其值回传给实参
- D) 实参将其值传递给形参，调用结束时形参并不将其值回传给实参

70、以下叙述中，不正确的是 B

- A) 在同一 C 程序文件中，不同函数中可以使用同名变量
- B) 在 main 函数体内定义的变量是全局变量
- C) 形参是局部变量，函数调用完成即失去意义
- D) 若同一文件中全局变量和局部变量同名，则全局变量在局部变量作用范围内不起作用

71、对于 C 语言的函数，下列叙述中正确的是 A

- A) 函数的定义不能嵌套，但函数调用可以嵌套
- B) 函数的定义可以嵌套，但函数调用不能嵌套
- C) 函数的定义和调用都不能嵌套
- D) 函数的定义和调用都可以嵌套

72、数组定义为：int a[4][5]; 引用“a[1]+3”表示的是 A

- (A) a 数组第 2 行第 4 列元素的地址
- (B) a 数组第 2 行第 4 列元素的值
- (C) a 数组第 4 行的首地址
- (D) a 数组第 1 行第 3 个元素的地址

73、若定义：int a=511,*b=&a; 则 printf("%d\n",*b); 的输出结果为 D

- A. 确定值
- B) a 的地址
- C) 512
- D) 511

74、若有 int a[10]={1,2,3,4,5,6,7,8,9,10};int *p=a; 则值为 9 的表达式是 B

- A. *p+9
- B. *(p+8)
- C. *p+=9
- D. p+8

75、若有说明语句：int a,b,c,*d=&c ; , 则能正确从键盘读入三个整数分别赋给变量 a、b、c 的语句是 A。

- A) scanf("%d%d%d",&a,&b,d);
- B) scanf("%d%d%d",&a,&b,&d);
- C) scanf("%d%d%d",a,b,d);
- D) scanf("%d%d%d",a,b,*d);

76、设有语句 int a[]={1,3,5,7,9},*p=&a[0]; 则值为 7 的表达式是 C

- A. *p+3
- B. *p+4
- C. *(p+3)
- D. *(p+4)

77、已知赋值语句 wang.year=2004; 则 wang 的变量类型是 C

- A. 字符或文件
- B. 整型或实型
- C. 共用或结构
- D. 实型或指针

78、设有以下说明语句：

struct stu

{ int a; float b; } stutype;

则下面的叙述不正确的是 C

- A . struct 是结构体类型的关键字
- B . struct stu 是用户定义的结构体类型名
- C . stutype 是用户定义的结构体类型名
- D. a 和 b 都是结构体成员名

79 、 下列关于结构的语法错误的是 D

- A . 结构是由用户自定义的一种数据类型
- B . 结构中可设定若干个不同数据类型的成员
- C . 结构中成员的数据类型可以是另一个已定义的结构
- D . 在定义结构时 , 可以为成员设置默认值

80 、 说明一个结构体变量时 , 系统分配给它的内存是 A

- A . 各成员所需内存量的总和
- B . 结构中第一个成员所需内存量
- C . 成员中占内存量最大者所需的容量
- D . 结构中最后一个成员所需内存量

三、 程序理解

1 . 以下程序运行结果是

a=5 , b=?
c=5 , d=5

```
#include <stdio.h>

int main()
{int a,b,c,d;

a=b=5;

c=(a++)+(a--)-(a++);

d=(b++)-(++b)+(--a);

printf("a=%d,b=%d\n",a,b);

printf("c=%d,d=%d\n",c,d);

return 0;

}
```

2 . 以下程序运行结果是

a=2 , b=2

```
#include <stdio.h>

int main()
{int x=1,a=0,b=0;

switch(x)
{
```

```

    case 0: b++;
    case 1: b=++a;
    case 2: a++;b++;break;
    case 3: ++a;++b;
    }
    printf("a=%d,b=%d\n",a,b);
return 0;
}

```

3 . 以下程序运行结果是

2,1

```

#include <stdio.h>

int main()
{int a,b,c=246;
a=c/100%9;
b=(1)&&(-1)||0;
printf("%d,%d\n",a,b);
return 0;
}

```

4 . 以下程序运行结果是

0,0,3

```

#include <stdio.h>

int main()
{int a=-1, b=4, k;
k=(++a<=0)&&(b--<=0);
printf("%d,%d,%d\n",k,a,b);
return 0;
}

```

5 . 以下程序运行结果是

60*5=300

```

#include <stdio.h>

int main()
{int a=5,b=60,c;
if (a<b)
{c=a*b;printf("%d*%d=%d\n",b,a,c);}
else
{c=b/a;printf("%d/%d=%d\n",b,a,c);}
return 0;
}

```

}

6 . 以下程序运行结果是 **5/6 0=0**

```
#include <stdio.h>

int main()
{int a=60,b=5,c;
if (a<b)
{c=a*b;printf("%d*%d=%d\n",b,a,c);}
else
{c=b/a;printf("%d/%d=%d\n",b,a,c);}
return 0;
}
```

7 . 以下程序运行结果是 **2,8**

```
#include <stdio.h>

int main()
{int a=1,b=10;
do
{ b-=a;a++;
} while(b--<0);
printf("%d,%d\n",a,b);
return 0;
}
```

8 . 以下程序运行结果是 **7 4**

```
#include <stdio.h>

int main()
{int x;
for(x=10;x>3;x--)
{ if(x%3) x--;
--x; --x;
printf("%d ",x);
}
return 0;
}
```

9 . 以下程序运行结果是 **ABA BA**

```
#include <stdio.h>
```

```

int main()
{
    int i;
    for(i=1;i<6;i++)
    {
        if (i%2!=0) {printf("A");continue;}
        printf("B");
    }
    printf("\n");
    return 0;
}

```

10 . 以下程序运行结果是

BA

```

#include <stdio.h>

int main()
{
    int i;
    for(i=1;i<6;i++)
    {
        if (i%2==0) {printf("A");break;}
        printf("B");
    }
    printf("\n");
    return 0;
}

```

11 . 以下程序运行结果是

1 5 9

```

#include <stdio.h>

int main()
{
    int i,x[3][3]={1,2,3,4,5,6,7,8,9};
    for(i=0;i<3;i++)
    {
        printf("%d ",x[i][i]);
    }
    printf("\n");
    return 0;
}

```

12 . 以下程序运行结果是

1 3 7 15

```

#include <stdio.h>

int main()
{
    int i,n[]={0,0,0,0,0};
    for(i=1;i<=4;i++)
    {
        n[i]=n[i-1]*2+1;
    }
}

```

```
printf("%d ",n[i]);
}
return 0;
}
```

13 . 以下程序运行结果是

s=54321

```
#include <stdio.h>

int main()
{int a[ ]={5,4,3,2,1},i,j;
long s=0;
for(i=0;i<5;i++)          s=s*10+a[i];
printf("s=%ld\n",s);
return 0;
}
```

14 . 以下程序运行结果是

abc

```
#include <stdio.h>

int main()
{char s[]="abcdef";
s[3]='\0';
printf("%s\n",s);
return 0;
}
```

15 . 以下程序运行结果是

14

```
#include <stdio.h>
#include <string.h>

int main()
{int a[3][3]={{1,2},{3,4},{5,6}},i,j,s=0;
for(i=0;i<3;i++)
for(j=0;j<i;j++)
s+=a[i][j];
printf("%d\n",s);
return 0;
}
```

16 . 以下程序运行结果是

xyabcABC

```
#include <stdio.h>
```

```
#include    <string.h>

int main()
{char p1[10]="abc", p2[10]="ABC", str[50]="xyz";
strcpy(str+2,strcat(p1,p2));
printf("%s\n",str);
return 0;
}
```

17. 以下程序运行结果是 31

```
#include <stdio.h>

int fun(int x,int y,int z)
{ z=x*x+y*y;}

int main()
{ int a=31;
  fun(5,2,a);
  printf("%d\n",a);
  return 0; }
```

18. 以下程序运行结果是 29

```
#include <stdio.h>

int fun(int x,int y,int z)
{ z=x*x+y*y;return z;}

int main()
{ int a=31;
  a=fun(5,2,a);
  printf("%d\n",a);
  return 0; }
```

19. 以下程序运行结果是

14

```
#include <stdio.h>

f(int b[ ], int n)
{ int i, r=0;
  for(i=0; i<=n; i++) r=r+b[i];
  return r;
}

int main()
{
  int x, a[]={ 2,3,4,5,6,7,8,9};
  x=f(a, 3);
  printf("%d\n",x);
  return;
}
```

20. 以下程序运行结果是

120

```
#include <stdio.h>

f(int b[ ], int n)
{ int i, r=1;
  for(i=0; i<=n; i++) r=r*b[i];
  return r;
}

int main()
{
  int x, a[]={ 2,3,4,5,6,7,8,9};
  x=f(a, 3);
  printf("%d\n",x);
  return;
}
```

21. 以下程序运行结果是

15

```
int func(int a,int b)
{ return(a+b); }
```

```

int main()
{
    int x=2,y=5,z=8,r;
    r=func(func(x,y),z);
    printf("%d\n",r);
    return 0;
}

```

22. 以下程序运行结果是 6

```

#include<stdio.h>

int f(char s[ ])
{
    int i,j;
    i=0;j=0;
    while( s[j]!='\0') j++;
    return (j-i);
}

int main()
{
    printf("%d\n",f("ABCDEF"));
    return 0;
}

```

23. 以下程序运行结果是 5

```

#include <stdio.h>

int f(int a)
{
    int b=0,c;
    c=3;
    b++ ;    c++;
    return (a+b+c);
}

int main()
{
    int i;
    for(i=0;i<3;i++)
        printf("%d\n",f(i));
    return 0;
}

```

24. 以下程序运行结果是 579


```
#include <stdio.h>

int f(int    a)
{ int b=0;
    static c=3;
    b++ ;    c++;
    return (a+b+c);    }

int main()
{ int i;
    for(i=0;i<3;i++)
        printf("%d\n",f(i));
    return 0;
}
```

25. 以下程序运行结果是 6

```
#include <stdio.h>

int fun3(int x)
{static int a=3;
    a+=x;
    return(a);    }

void main()
{int k=2,m=1,n;
    n=fun3(k);
    n=fun3(m);
    printf("%d\n",n);    }
```

26. 以下程序运行结果是 2

```
#include <stdio.h>

int m=13;

int fun2(int x, int y)
{ int m=3;
    return(x*y-m);
}

int main()
{ int a=7, b=5;
    printf("%d\n",fun2(a,b)/m);
}
```

```

    return 0;
}

```

27. 以下程序运行结果是

```

a=100,b=10
*pointer_1=100,*pointer_2=10

```

```

#include <stdio.h>

int main()
{ int a=100,b=10;
  int *pointer_1, *pointer_2;
  pointer_1=&a;
  pointer_2=&b;
  printf("a=%d,b=%d\n",a,b);
  printf("*pointer_1=%d,*pointer_2=%d\n",*pointer_1,*pointer_2);
  return 0;
}

```

28. 以下程序运行结果是

```

a=11,b=121
max=121,min=11

```

```

#include <stdio.h>

int main()
{ int *p1,*p2,*p,a,b;
  a=11,b=121;
  p1=&a;
  p2=&b;
  if(a<b)
  {p=p1;p1=p2;p2=p;}
  printf("a=%d,b=%d\n",a,b);
  printf("max=%d,min=%d\n",*p1,*p2);
  return 0;
}

```

29. 以下程序运行结果是

```

1 2
3 4
5 6

```

```

#include<stdio.h>

void main()
{
  int i,j,(*p)[2];

```

```
int b[3][2] = {{1, 2}, {3, 4}, {5, 6}};

p = b;
for(i=0; i<3; i++)
{for(j=0; j<2; j++)
    printf("%d ",p[i][j]);
printf("\n");}
}
```

30. 以下程序运行结果是

1	-2	9	
4	7	0	10

```
#include<stdio.h>

void main()
{
    int i,*p[2];
    int a[3] = {1, -2, 9};
    int b[4] = {4, 7, 0, 10};
    p[0]=a;
    p[1]=b;
    for(i = 0; i < 3; i++)
printf("%d\t", *(p[0]+i));

    printf("\n");
    for(i = 0; i < 4; i++)
printf("%d\t", *(p[1]+i));

    printf("\n");}
```

31. 以下程序运行结果是

NO.:10101
name:Li Lin
sex:M
address:123 Beijing Road

```
#include <stdio.h>

int main()
{struct student
    {long int num;
    char name[20];
    char sex;
    char addr[20];
    }a={10101,"Li Lin",'M',"123 Beijing Road"};

printf("NO.:%ld\nname:%s\nsex:%c\naddress:%s\n",a.num,a.name,a.sex,a.addr
);
```

```
return 0;
}
```

No.	Name	sex	age
10101	Li Lin	M	18
10102	Zhang Fun	M	19
10104	Wang Min	F	20

32. 以下程序运行结果是

```
#include <stdio.h>

struct student
{
    int num;

    char name[20];

    char sex;

    int age;
};

struct student stu[3]={10101,"Li Lin",'M',18},{10102,"Zhang Fun",'M',19},
                        {10104,"Wang Min",'F',20}};

int main()
{
    struct student *p;

    printf(" No.      Name                      sex age\n");

    for (p=stu;p<stu+3;p++)

        printf("%5d %-20s %2c %4d\n",p->num, p->name, p->sex, p->age);

    return 0;
}
```

四、编程

1 . 设有一个函数如下。编程，输入 x，求 y 值。

$$y=\begin{cases} \sqrt{x} & x>0 \\ 0 & x=0 \\ \frac{x+1}{x^2+2} & x<0 \end{cases}$$

```
#include<stdio.h>

#include<math.h>

int main()
{
    float x,y;
```

```
scanf("%f",&x);

if (x>0)
    y=sqrt(x);
else if(x<0)
    y=(x+1)/(x*x+2);
else
    y=0;

printf("x=%6.2f,y=%6.2f\n",x,y);

return 0;

}
```

2 . 下面程序的功能是计算 $1-3+5-7+\dots-99+101$ 的值存在 s 中。

```
#include <stdio.h>

int main()
{
    int i,t=1,s=0;
    for(i=1;i<=101;i+=2)
        {
            s=s+i*t;
            t=-t;
        }
    printf("sum=%d\n", s);
    return 0;
}
```

```
#include <stdio.h>

int main()
{
    int i=1,t=1,s=0;
    while(i<=101)
        {
            s=s+i*t;
            t=-t;
            i+=2;
        }
    printf("sum=%d\n", s);
    return 0;
}
```

```
#include <stdio.h>

int main()
{
    int i=1,t=1,s=0;
    do
        {
            s=s+i*t;
            t=-t;
            i+=2;
        } while(i<=101);
    printf("sum=%d\n", s);
    return 0;
}
```

3 . 一个整数被 5 除余 3 , 被 7 除余 5 , 被 9 除余 7 , 这个数最小是几 ?

```
#include<stdio.h>

int main()
{int i=8;
do
{if ((i%5==3) && (i%7==5) && (i%9==7))
{printf("%5d\n",i);
break;}
i=i+5;} while(1);
return 0;
}
```

4. 编写一程序 , 将两个字符串连接起来 , 要求不用 `strcat` 函数。

```
#include<stdio.h>
#include<string.h>
void main()
{ char s1[80],s2[40];
int i=0,j=0;
printf("input string1:\n");
gets(s1);
printf("input string2:\n");
gets(s2);
i=strlen(s1);
while( s2[j]!='\0')
s1[i++]=s2[j++];
s1[i]='\0';
puts(s1);
}
```

5 . 写一个函数计算 2016 以内最大的 10 个能被 13 或 17 整除的自然数之和。

输出格式 : `sum=23456`

```
#include<stdio.h>

void main()
{
long sum;
```

```

sum=sum2016();
printf("sum=%ld\n",sum);
}
long sum2016()
{
long sum=0;
int i,count=0;
for(i=2016;;i--)
{ if(i%13==0||i%17==0)
{count++;
sum+=i;}
if(count==10) break;}
return sum;
}

```

6 . 已知圆柱高 $h=5$, 要求用 scanf 输入圆的半径 , 求出圆周长 , 圆面积和圆柱体积。

```

#include <stdio.h>
int main ()
{float h=5,r,l,s,vz;
float pi=3.1415926;
printf(" 请输入圆半径 r:");
scanf("%f",&r); // 要求输入圆半径 r
l=2*pi*r; // 计算圆周长 l
s=r*r*pi; // 计算圆面积 s
vz=pi*r*r*h; // 计算圆柱体积 vz
printf(" 圆周长为 : l=%6.2f\n",l);
printf(" 圆面积为 : s=%6.2f\n",s);
printf(" 圆柱体积为 : vz=%6.2f\n",vz);
return 0;
}

```

7. 试编写程序输出 100 ~ 300 间的所能同时被 3 和 5 整除的数。

```

#include <stdio.h>
int main()
{int i;

```

```

for(i=100;i<=300;i++)
if((i%3==0)&&(i%5==0))
printf("%5d",i);
return 0;
}

```

8 . 一个球从 100m 高度自由落下，每次落地后反跳回原高度的一半，再落下，再反弹。求它在第 10 次落地时，共经过多少米，第 10 次反弹多高。

```

#include <stdio.h>

int main()
{
    double sn=100,hn=sn/2;
    int n;
    for (n=2;n<=10;n++)
    {
        sn=sn+2*hn;          /* 第 n 次落地时共经过的米数 */
        hn=hn/2;             /* 第 n 次反跳高度 */
    }
    printf(" 第 10 次落地时共经过  %f  米\n",sn);
    printf(" 第 10 次反弹  %f  米\n",hn);
    return 0;
}

```

9 . 以下程序从终端读入数据到数组中，统计其中偶数的个数，并计算它们之和及平均值。

```

#include <stdio.h>

int main()
{ int i,a[20],count;float ave,sum;
  sum=0.0;count=0;ave=0;
  for(i=0;i<20;i++) scanf("%d", &a[i] );
  for(i=0;i<20;i++)
  { if( a[i]%2==0 )
    { count++;
      sum=sum+a[i];
    }
  }
}

```



```

ave=sum/count;

printf("sum=%f,count=%d,ave=%.2f\n",sum,count,ave);

return 0;

}

```

10 . 水仙花数的判定。

“水仙花数”是指一个三位数，其各位数字立方和等于该数本身。

例如：153 是一个水仙花数，因为 $1^3+5^3+3^3=153$ 。

```

#include <stdio.h>

int main()
{
    int num,i,j,k;

    do
    {printf("    请输入一个三位数：    \n");

    scanf("%d",&num);

    }while (num>999||num<100);

    i=num/100;

    j=num/10%10;

    k=num%10;

    printf("    您输入的数是 %d\n    百十个位数字依次为：    %d,%d,%d\n",num,i,j,k);

    if(i*i*i+j*j*j+k*k*k==num)

    printf("%d    是水仙花数 \n",num);

    else

    printf("%d    不是水仙花数 \n",num);

    return 0;

}

```

11 . 给定一 3*4 的矩阵，求出其中的最大元素值，及其所在的行列号。

```

#include <stdio.h>

int main()

{int i,j,row=0,colum=0,max;

    int a[3][4]={{1,2,3,4},{9,8,7,6},{-10,10,-4,4}};                // 定义数组并赋初值

    max=a[0][0];                                                    // 先认为 a[0][0] 最大

    for (i=0;i<=2;i++)

        for (j=0;j<=3;j++)

```

```

        if (a[i][j]>max)                // 如果某元素大于 max , 就取代
max  的原值
        {max=a[i][j];
          row=i;                        // 记下此元素的行号
          colum=j;                     // 记下此元素的列号
        }
printf("max=%d\nrow=%d\ncolum=%d\n",max,row,colum);
return 0;
}

```

12. 矩阵转置：将二维数组 a 的行和列元素互换后存到另一个二维数组 b 中。

```

#include <stdio.h>

int main()
{ int i,j,b[3][2],a[2][3]={1,2,3},{4,5,6}};

printf("array a :\n");
for(i=0;i<2;i++)
{for(j=0;    j<3    ;j++)
    {printf("%5d",a[i][j]);
      b[j][i]=a[i][j] ;      }
  printf("\n");
}

printf("array b :\n");
for(i=0;    i<3    ;i++)
{for(j=0;j<=1;j++)
    printf("%5d",b[i][j]);
  printf("\n");
}

return 0;}

```

13 . 用数组来处理求 Fibonacci 数列问题。输出前 20 个数，每行 5 个。

```

#include <stdio.h>

int main(    )
{
int    i;
    int f[20]={1,1};
for(i=2; i<20; i++)

```

```

        f[i]=f[i-1]+f[i-2];
for(i=0; i<20; i++)
{
if(i%5==0) printf("\n");
printf("%7d" , f[i]);
}
printf("\n");
return 0;
}

```

14. 输出杨辉三角。

```

#include <stdio.h>

#define N 7

int main()
{ int i,j,a[N][N];
  for (i=0;i<N;i++)
  {a[i][i]=1;
   a[i][0]=1;
  }
  for (i=2;i<N;i++)
  for (j=1;j<=i-1;j++)
    a[i][j]=a[i-1][j-1]+a[i-1][j];
  for (i=0;i<N;i++)
  {for (j=0;j<=i;j++)
    printf("%6d",a[i][j]);
    printf("\n");
  }
  printf("\n");
  return 0;
}

```

15. 输入一个年份，判断其是否为闰年，并输出判断结果。

```

#include <stdio.h>

int main()
{

```

```

int year, leap;

scanf("%d", &year);

if(year%4!=0)
    leap=0;
else if (year%100!=0)
    leap=1;
else if(year%400!=0)
    leap=0;
else
    leap=1;
if (leap)
    printf("%d is ", year);
else
    printf("%d is not ", year);
printf("a leap year.\n");
return 0;
}

```

16. 写一个函数，输出九九乘法表。

```

#include <stdio.h>

void multi()
{
    int i, j, m=0;
    for(i=1; i<=9; i++)
    {
        for(j=1; j<=i; j++)
        {
            printf("%d*%d=%d\t", i, j, i*j);
        }
        printf("\n");
    }
}

int main()
{
    multi();
    return 0;
}

```

17. 写一个判断素数的函数，在主函数输入一个整数，输出是否为素数的信息。

```

#include <stdio.h>

int main()
{
    int prime(int);
    int n;
    printf("input an integer:");
    scanf("%d",&n);
    if (prime(n))
        printf("%d is a prime.\n",n);
    else
        printf("%d is not a prime.\n",n);
    return 0;
}

int prime(int n)
{
    int flag=1,i;
    for (i=2;i<n/2 && flag==1;i++)
        if (n%i==0)
            flag=0;
    return(flag);
}

```

18. 写一个增加字符间空格的函数，在主函数输入 4 个字符，输出时每两个字符间留一个空格。
如输入 aB16 ，应输出 a B 1 6 。

```

#include <stdio.h>
#include <string.h>

int main()
{
    char str[80];
    void insert(char []);
    printf("input four digits:");
    scanf("%s",str);
    insert(str);
    return 0;
}

void insert(char str[])
{
    int i;
    for (i=strlen(str);i>0;i--)

```

```
    {str[2*i]=str[i];  
      str[2*i-1]=' '  
    }  
    printf("output:\n%s\n",str);  
}
```

19. 求两个整数的最大公约数与最小公倍数。

```
#include <stdio.h>  
  
int main()  
{int hcf(int,int);  
  int lcd(int,int,int);  
  int u,v,h,l;  
  scanf("%d,%d",&u,&v);  
  h=hcf(u,v);  
  printf("H.C.F=%d\n",h);  
  l=lcd(u,v,h);  
  printf("L.C.D=%d\n",l);  
  return 0;  
}  
  
int hcf(int u,int v)  
{int t,r;  
  if (v>u)  
    {t=u;u=v;v=t;}  
  while ((r=u%v)!=0)  
    {u=v;  
      v=r;}  
  return(v);  
}  
  
int lcd(int u,int v,int h)  
{  return(u*v/h);  }
```